SOFTWARE ARTS TECHNICAL NOTES, BY THE CREATORS OF VISICALC

A Publication of
Software Arts, Inc.

SATN

THE JOURNAL FOR VISICALC USERS

Vol. II, No. 1
September/October, 1982

IN THIS ISSUE:

About This Issue 1
VisiCalc and VisiPlot 1
From Our Readers 5
ACRS Depreciation 6
A Look at VisiCalc Advanced Version 11
VisiCalc Related Publications 12

ABOUT THIS ISSUE

We've received a number of calls from readers asking for a clearer statement of policy regarding their freedom to copy the Software Arts Spreadsheet Planner that appeared in the last issue of SATN. We will raise no objections if a subscriber makes copies of the Software Arts Spreadsheet Planner for his or her personal use or for use in his or her business. We do claim copyright of the Spreadsheet Planner, though, and will object to its reproduction for sale or for distribution to non-subscribers.

In a survey of almost 4000 SATN readers (about which we'll report briefly in the next issue) we learned that a lot of you use the VisiPlot™ program. Mitchell Kapor, the author of the VisiPlot/VisiTrend™ program, has written an article for us showing how to use DIF™ to exchange data between the VisiCalc® and VisiPlot programs. Mitch's article contains a number of useful hints about DIF, and if you are among the nearly one in four SATN readers who regularly use the VisiPlot program, you should find the article of real value. In upcoming issues we hope to bring you more articles about using DIF to communicate with other programs.

Several readers have asked how to create a table in which various calculations are distributed based on the years or months to which they refer. One way to do so is shown in *ACRS Depreciation*. Although depreciation under the new tax law is its main example, the article is really about how to rewrite tables so that you can get the VisiCalc program to select the information you need from as many as a dozen different ones. What we've done won't work all the time in all applications, but the article does contain a number of techniques that can be applied in a variety of business situations other than analyzing depreciation. The functions highlighted in the article include @IF, @AND, @CHOOSE, and @LOOKUP.

In *From Our Readers* a reader shows us the expression to use if you want to calculate the interest portion of a loan payment according to the "rule of 78". This calculation is the one that banks use widely for short-term loans, like car loans, and is the same as the expression used for sum of the years' digits depreciation.

The VisiCalc Advanced Version program is a reality now, and although initially it will be available only for the Apple ///, it will be available on other machines in the future. There is much about VisiCalc Advanced Version that is familiar to current users of the program, but there is a great deal that is new, too. In *A Look at VisiCalc Advanced Version* we've adapted some of the figures in other articles to Advanced Version, and discuss some of the exciting new formatting capabilities of VisiCalc Advanced Version.

Complementing the hardware/software listings in the last issue of SATN is this issue's listing of books and articles about using the VisiCalc program.

VISICALC AND VISIPLOT

by Mitchell Kapor

It is often easier to grasp relationships between columns of figures if they can be presented in graphic as well as numerical terms. The DIF™ format, created by Software Arts, Inc. to permit data to be transferred between microcomputer applications programs, allows you to load data from your VisiCalc® worksheets into a graphing package like the VisiPlot™ program and easily make graphs and charts of it.

This article explains how to use the DIF format to transfer data from the VisiCalc program to the VisiPlot and VisiTrend™/VisiPlot programs and includes some tips on preparing your VisiCalc sheets to transfer them more efficiently. (The VisiTrend program includes the complete VisiPlot program and adds analytical and statistical procedures.) Unless specifically noted, everything we say about VisiPlot in this article applies equally to VisiTrend.

KEY TO CONVENTIONS USED IN SATN

The VisiCalc® program is available on personal computers manufactured by Apple, Atari, Commodore, Hewlett-Packard, IBM, and Radio Shack. To reduce any confusion arising from the differences among the keyboards of those computer systems, SATN uses a standard symbol and name to represent certain commonly used keys.

The symbol ® and the word RETURN are used as synonyms for the keys known as ENTER, ENDLINE, and RETURN. The symbol ^ is used to represent an exponentiation operator.

To make VisiCalc documentation more readable, the program was designed so that the colon (:) has the same effect as RETURN in some commands, as when ending the > command or separating various elements of the replicate command.

In examples of what to type, type exactly what you see and press RETURN at the end of every line. Users of the HP-83/85 may find it necessary to add parentheses to some of the expressions included in the text. For example, 3 + 4 ∗ 5 should be (3 + 4) ∗ 5 on an HP-83/85.

Throughout SATN, one typeface represents both VisiCalc input and output (what you should type and what you will see on the screen). The typeface looks like this:

```
This line is typeset in OCR-B.
So is this: +A1+B1*3.25
```

To allow you to type the formulas directly on a VisiCalc sheet, examples in SATN specify coordinates, like A1, rather than the names of the values they represent. When actually typing a formula, you can use the arrows to point rather than using the > command.

SATN is published bi-monthly by Software Arts, Inc., 27 Mica Lane, Wellesley, Massachusetts 02181. Address all editorial correspondence to the Editorial Department, SATN, 27 Mica Lane, Wellesley, MA 02181. Requests for special permission or bulk orders should be addressed to the publisher.

Domestic subscriptions: 6 issues, $30.00; outside U.S.A., $50.00. Single copy price, $6.00. Send subscriptions, change of address, and fulfillment questions to SATN Subscriptions, P.O. Box 815, Quincy, MA 02169.

Copyright © 1982, Software Arts, Inc. All rights reserved. This periodical may not be copied or reproduced in any fashion either in whole or in part without the prior written consent of Software Arts, Inc.

Software Arts, Inc. makes no warranty, express or implied, with respect to the quality, accuracy or freedom from error of the information contained in SATN, the journal for VisiCalc users, and specifically disclaims all liability for any damages resulting from the use of such information.

Unless specifically stated otherwise in any article in this issue of SATN, Software Arts, Inc. has not examined any products and/or publications identified in SATN to determine their accuracy and/or ability to perform as advertised. Mention of a product and/or publication in SATN does not constitute an endorsement by Software Arts, Inc. of any such product or publication.

Apple is a registered trademark of App
Atari Inc. Commodore is a trademar
trademark of Software Arts, Inc. HP-83/85 and HP-125 are trademarks of Hewlett-Packard Company. IBM is a registered trademark of International Business Machines Corporation. TRS-80 is a trademark of Radio Shack Division of Tandy Corp.

SATN is a trademark of Software Arts, Inc. VisiCalc is a registered trademark of VisiCorp VisiPlot and VisiTrend are trademarks of VisiCorp

How the VisiCalc and VisiPlot Programs See the World

The VisiCalc program sees the world as a collection of interrelated labels, numbers, and formulas. When you create a DIF file, you save a single, rectangular portion of the sheet as data. That is, you write into the file the labels, numbers, and actual values of expressions, but not the expressions themselves.

The VisiPlot program sees the world as a collection of time-series. Your company's annual sales for five years is a time-series. So is the weekly closing price of a particular stock. A time-series, which we'll call a series, is a list of numbers with associated dates. More formally, a series is characterized by a number of data points and their date information, which consists of a periodicity and a start year. A periodicity is the number of data points per year. An annual series has periodicity 1, which is to say there is one data point for each year. A monthly series, then, has periodicity 12. A start year is the first year in the series.

For example, suppose your company's annual sales are:

1977	300,000	1979	335,000
1978	325,000	1980	350,000

In VisiPlot terminology, this series has a periodicity 1, a start year 1977, and four data points.

Or suppose the monthly closing price of your favorite computer stock is:

Feb 1981	53.00	May 1981	70.00
Mar 1981	59.25	Jun 1981	72.00
Apr 1981	62.50	Jul 1981	73.25

That series has periodicity 12 (monthly), a start year of 1981, a start period of 2, and six data points. When a series' periodicity is not annual, that is, not 1, its start date consists of a start year and a start period (i.e., January = 1, February = 2, etc. for monthly series).

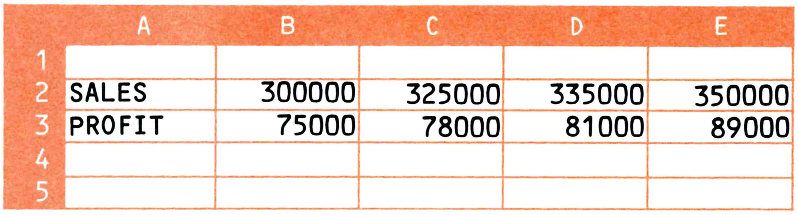
Transferring a VisiCalc Sheet to the VisiPlot Program

When you transfer data using the DIF format, each row (or column) on the VisiCalc sheet becomes one series in the VisiPlot program. Assuming you select row order, saving two or more rows of the sheet creates the same number of series in the VisiPlot program. If you save a DIF file consisting of two rows that extend across six columns, you create two VisiPlot series with six data points each. However, if you save the same file in column order you will create six VisiPlot series with two data points each.

Suppose, for example, you are working on the VisiCalc sheet shown in Figure 1. (We'll assume that you are using two drives, though by swapping disks, you can use only one.) To create a DIF file you could plot on the VisiPlot program, you would enter the following:

```
>A2:/S#S
```

The program will prompt for a filename. Call the file DIFTEST1. After entering the filename, you will be prompted again for the lower right of the screen to define the portion of the sheet you wish to save. You may either point with the cursor or type the entry E4.



	A	B	C	D	E
1					
2	SALES	300000	325000	335000	350000
3	PROFIT	75000	78000	81000	89000
4					
5					

FIGURE 1. *The DIF file to be transferred to the VisiPlot program begins at A2. Including the first row is unnecessary, since you can specify the year the series begins when you have loaded the VisiPlot program.*

The final prompt reads: DATA SAVE: R, C OR RETURN (or ENTER on the IBM version of the VisiCalc program). If you wish to make each row into a series, then you must save the file in ROW order. Since ROW order is the default, enter R or ®.

You are now ready to read the data into the VisiPlot program. Remove the VisiCalc program diskette and insert the VisiPlot program diskette into Drive 1. Assuming that your disk controller is in slot 6, quit the VisiCalc program and boot the VisiPlot program by entering:

```
/SQ6
```

Once the Visiplot program has booted, a menu will appear at the top of the screen with the cursor pointing to the command LOAD. Press ® and the message READING DIRECTORY...will replace the menu. A vertical list of all the files on your data diskette, including DIFTEST1, will appear below.

Using the arrow keys, move the cursor to DIFTEST1 and press ®. After a few seconds, the prompt PERIODICITY will appear. On the same line it will also say CURRENT : 1. Now you have to tell VisiPlot the periodicity of the series. Since the series in the example has a periodicity 1, and that is the current periodicity, you need only type ®.

A new prompt will appear: MAJOR START (YEAR)? Since 1977 is the year that begins the series, enter 1977. As the rest of the file is read in, you will see the message PLEASE WAIT...

After a few seconds, the following display will appear on the screen below the status area:

```
NAME     PER      START     END       #

SALES     1        1977     1980      4
PROFIT    1        1977     1980      4
```

The data is now available for you to use in the VisiPlot program.

Once you have mastered the basic operation of saving a DIF file and loading it in the VisiPlot program, you can move on to some refinements that can make your applications more effective.

Handling Labels

For your own convenience, the first column of a DIF file should, if possible, consist entirely of labels or entirely of values.

When the VisiPlot program looks at a DIF file, it reads the entry in the upper left corner of the file first. If that entry is a label, all the other entries in the column are assumed to be labels, too, as in Figure 1. Each label then becomes a series name. If there are values in the first column, and the column is headed by a label, and the column has been saved by row, the VisiPlot program will give you the error message BAD DIF FORMAT, even though you are trying to load a valid DIF file.

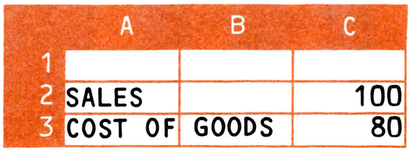
If the first column in the file that you save begins with a value and then contains one or more labels, then all the entries in the column are treated as data. That is, none of the labels will be used as a series name. Instead the program will assign series names to the entries: SERIES0, SERIES1, SERIES2, and so on.

Blank Entries

Although you may often save DIF files that contain blank rows or columns, those blanks will be interpreted by the VisiPlot program either as data points of zero or labels of zero length. That can produce some unexpectedly odd looking graphs. The best strategy to adopt is to save the VisiCalc sheet with the /SS command first, then delete any blank rows or columns using the /DR and /DC commands before saving your DIF file.

A Tip on Multicolumn Labels

To avoid cryptic labels, it's sometimes necessary to use more than one column. If, as in the VisiCalc sheet shown in Figure 2, you have set your column width to 7, the usual approach to the label COST OF GOODS would be to enter COST OF in column A and GOODS in column B. To avoid the run-on display, COST OFGOODS, though, you must begin the second part of the label with a leading blank. Moreover, if you also include a shorter label, SALES, for example, you will have a blank entry in B2, which will be interpreted as a zero data point or a zero length label.



	A	B	C
1			
2	SALES		100
3	COST OF	GOODS	80

FIGURE 2. *If the label in A3 is made to read COST OF GOODS SOLD, and column B is deleted before making a DIF file to be transferred to VisiPlot, your DIF file will still contain the entire label in A3, even though the screen will only show the label COST OF.*

If you then created a DIF file starting at A2, the VisiPlot program would create two series, called SALES and COST OF, based on the contents of the entries in column A. However, since column B contains a blank entry followed by the label GOODS, the first data point in each series would have a value of zero. It is likely that you wanted the series to begin with the data points 100 and 80, as in column C.

One way around the problem is to save the sheet using /SS, delete column B using /DC, and then saving a DIF file. But you would then have a series called COST OF. There is a better way.

Remember that although the VisiCalc program displays only the part of a label that fits within the column width you define, you can enter a much longer label on the edit line. By entering COST OF GOODS at A3 and GOODS (with a leading blank) in B3, you will create a slightly redundant file, but you can then delete column B before saving the sheet as a DIF file, and still retain the proper label when the file is read by the VisiPlot program. That is, A3 will contain the label COST OF GOODS, but only display COST OF on the screen. At B3, the label GOODS appears on the screen in full. Even after deleting column B to create the DIF file, the entire label remains intact in A3 and will appear in full when the VisiPlot program reads the file.

Leading blanks do not create the same problems as blank entries. If you use labels with leading blanks, the VisiPlot program creates a series name with a leading blank.

Format Information

DIF files do not include any format information. If you have formatted the entries on your VisiCalc sheet with /FI or /F$, they will be loaded in the VisiPlot program in an unformatted condition. For example, if an entry has the value 32.753216 and is formatted with /F$, it will be displayed as 32.75 on the VisiCalc sheet. In the DIF file the full precision will be recorded: 32.753216. When you create your VisiCalc sheet, of course, you may use the formula, shown in *Using @LOOKUP to Calculate Withholding Taxes* (SATN Vol I, No. 1: September/October, 1981), for rounding to the nearest penny, and thus calculate the value so that 32.753216 is rounded to 32.75. However, if that is not practical or desirable, the VisiTrend/VisiPlot programs have a formatting capability, so that you can set a format within the VisiPlot program that makes the data appear exactly as it does on the VisiCalc sheet.

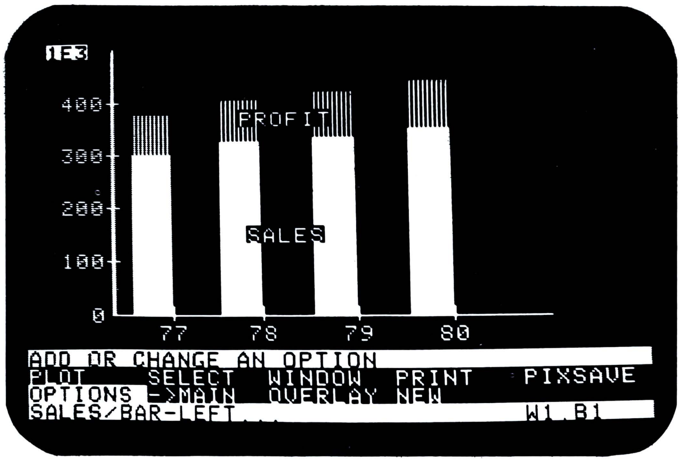
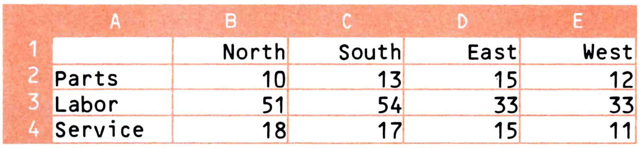


FIGURE 3. *This is a screen photo of the sheet shown in Figure 1.* *Photo by Sonny Tarbi.*

Cross Sectional Data

Sometimes, your data may not be time-series oriented, in either the VisiCalc or VisiPlot programs. Unfortunately, VisiPlot has no direct way to use data that is not time oriented, like cross-sectional data. We can use DIF to get around that problem, though.

Imagine a company with four regional service centers, (North, South, East, and West) each of which bills customers for parts, labor, and other services performed. A portion of a VisiCalc sheet might look like the one shown in Figure 4.



	A	B	C	D	E
1		North	South	East	West
2	Parts	10	13	15	12
3	Labor	51	54	33	33
4	Service	18	17	15	11

FIGURE 4. *The DIF file made from this sheet, and saved by row from A2 to E4, was used to generate the graphic shown in Figure 5.*

If we wanted to look at the service categories by region, we would save the sheet from A2...E4 (NOT saving the first row), and we would save by rows. When the DIF file is read into VisiPlot, we would assign each series a periodicity of 1 and a start year of 1, remembering that the first observation in each series corresponds to the region North, the second to South, and so forth. That would create the following series in VisiPlot:

```
Parts      10,  13,  15,  12
Labor      51,  54,  33,  33
Service    18,  17,  15,  11
```

We might want to make bar charts, for example, comparing the labor expenses in the regions. To do so, simply make a bar chart of the series PARTS in VisiPlot. We could even UNLABEL the X-axis and use moveable titles to identify the bars as regions.

Suppose, however, we wanted to look at the relative division of revenue for each region by category. That is, make a bar chart for the North region in which the first bar corresponds to Parts, the second to Labor, and the third to Services. We can do that by specifying the C (column) option when we save the DIF file. Then each row becomes a column and is saved as one VisiPlot series. If we saved the rectangle from B1..E4 (not saving column A), we would create the following VisiPlot series:

```
NORTH      10, 51, 18
SOUTH      13, 54, 17
EAST       15, 33, 15
WEST       12, 33, 11
```

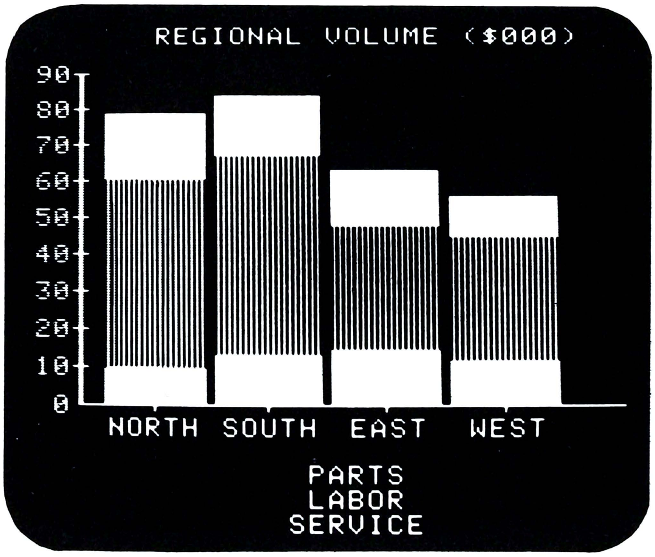


FIGURE 5. *This screen photo shows the bar graph made from the sheet in Figure 4. The legend at the top of the graph and the labels at the bottom were added from within the VisiPlot program.*

Photo by Sonny Tarbi.

The first data point in each series, which would correspond to the first bar in the graph, would be parts; the second data point and second bar would represent labor; and the third, service.

Mitchell Kapor is the author of the VisiPlot/VisiTrend program

BACK ISSUES AVAILABLE

Back issues of SATN are now available. The cost is $6.00 per copy for orders mailed within the USA. If your order is to be mailed outside the USA, add $2.50 for shipping and handling to the full price of the order.

To order back issues, please send your name and address, a list of the specific issues of SATN you want (by Volume and Issue numbers), and a check or money order payable to Software Arts, Inc., to:

SATN Back Issues
27 Mica Lane
Wellesley, MA 02181

FROM OUR READERS

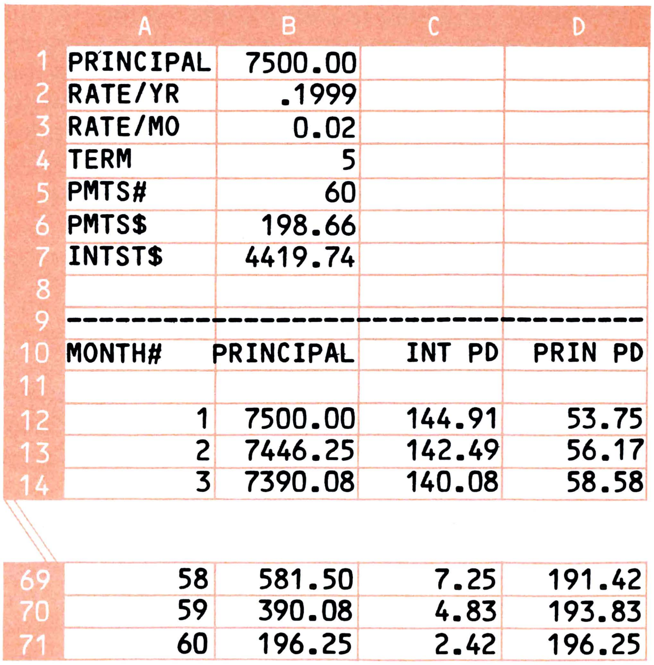
The article *Mortgages and Loans* in the November/December issue of SATN was helpful. But when I used your method to figure out how much interest I would pay on an auto loan, my numbers didn't quite jibe with my banker's. The total interest charged is the same, but the amount of interest in each payment differed.

It seems the interest part of an auto loan payment, at least here in Massachusetts, is calculated by something called the "rule of 78." It's the same as sum of the years' digits method of depreciation, only used for loans. The disk enclosed shows how it's done.

Stephen M. Chase, CLU
Wellesley, MA

Editor's Note: Using Steve's disk we produced the output for the sheet shown in Figure 6. The key expressions and the value references for replicating them into a table are listed below. The expression at C12, by the way, is precisely the one to use for the calculation of sum of the years' digits depreciation, as Steve points out. If you use his expression for depreciation calculations, simply substitute the depreciable value of the asset for the amount of interest.

```
/GOR
>B3:+B2/12
>B5:+B4*12
>B6:+B1*B3/(1-(1+B3^-B5))
>B7:+B5*B6-B1
>A12:1
>B12:+B1
>C12:/F$+B7*(B5-A12+1)/(B5+1*B5/2)
>D12:+B6-C12
>A13:+A12+1
>B13:+B12-D12
>C12:/R:...D12:C13:NNRNNNR
>A13:/R:...D13:A14...A71:RRRNNRNNNR
```



	A	B	C	D
1	PRINCIPAL	7500.00		
2	RATE/YR	.1999		
3	RATE/MO	0.02		
4	TERM	5		
5	PMTS#	60		
6	PMTS$	198.66		
7	INTST$	4419.74		
8				
9	----------	----------	----------	----------
10	MONTH#	PRINCIPAL	INT PD	PRIN PD
11				
12	1	7500.00	144.91	53.75
13	2	7446.25	142.49	56.17
14	3	7390.08	140.08	58.58
69	58	581.50	7.25	191.42
70	59	390.08	4.83	193.83
71	60	196.25	2.42	196.25

FIGURE 6. *The sheet for calculating interest and principal payments according to the "rule of 78."*

ACRS DEPRECIATION

Until 1981 calculating the best depreciation schedule was a matter of choosing one of three methods and then changing from one to another when doing so was financially advantageous and legally permissible. Under the so called Accelerated Cost Recovery System (ACRS) in the Economic Recovery Tax Act of 1981, such calculations have been made easier to perform, as the choices have been built into a set of schedules included in the Act.

Of course, if you developed VisiCalc® models to calculate depreciation according to the old methods, ACRS made it necessary to rewrite them. (ACRS hasn't been adopted by all states, so your models may still be useful for state tax purposes.) But if you have begun to rewrite your models, you have probably had to face the question: How do I get the VisiCalc program to choose the appropriate schedule?

As you can see from Table 1, the schedule you must use depends on whether the property is put in service between 1981 and 1984, in 1985, or after 1985; and whether its depreciable life is 3, 5, 10, or 15 years. Considering each class of property and the starting years of service, there are a dozen different schedules to deal with.

We have several aims in presenting this article. First, we wish to introduce @CHOOSE in more detail than has been given to it either in SATN or other publications. Second, we wish to deal with the problem of selecting the appropriate schedule from multiple ones. Third, as promised in earlier issues of SATN, we want to review the use of @IF and @AND. Fourth, we want to suggest a way for you to set up your own models so that they can be used by relatively untrained operators.

The sheet that we have created for those purposes allows us to perform depreciation calculations. Once entered into the memory of your personal computer, all the sheet requires of the user is that he or she supply three variables: the depreciable amount, the year the property is put in service, and the life of the property. The model itself finds the appropriate schedule, applies it to the depreciable amount, and distributes the results into a year by year table.

Rethinking the ACRS Schedules

Since the main problem is to select a schedule, it would seem at first blush that @LOOKUP should be used. But the problem doesn't yield easily to @LOOKUP, because that function expects a range that is in ascending order. If you work with the ACRS tables for a while, you soon find that there are many places where they are not in ascending order, and so do not make very usable @LOOKUP ranges.

The key problem is less a matter of which function to use than of rewriting the tables so that we can efficiently extract the information we need. The tables that Congress included in the Economic Recovery Tax Act are structured to be manipulated manually and emphasize the year that the property is placed in service. For that matter, most printed tables are structured for manual use and often require some sort of modification if they are to be used with the VisiCalc program.

TABLE 1
For Property Placed in Service 1981-84

	Apply This Percentage:			
Recovery Year	3 – Year	5 – Year	10 – Year	15 – Year*
1	25	15	8	5
2	38	22	14	10
3	37	21	12	9
4		21	10	8
5		21	10	7
6			10	7
7			9	6
8			9	6
9			9	6
10			9	6
11				6
12				6
13				6
14				6
15				6
For Property Placed in Service in 1985				
1	29	18	9	6
2	47	33	19	12
3	24	25	16	12
4		16	14	11
5		8	12	10
6			10	9
7			8	8
8			6	7
9			4	6
10			2	5
11				4
12				4
13				3
14				2
15				1
For Property Placed in Service After 1985				
1	33	20	10	7
2	45	32	18	12
3	22	24	16	12
4		16	14	11
5		8	12	10
6			10	9
7			8	8
8			6	7
9			4	6
10			2	5
11				4
12				3
13				3
14				2
15				1

*Public Utility

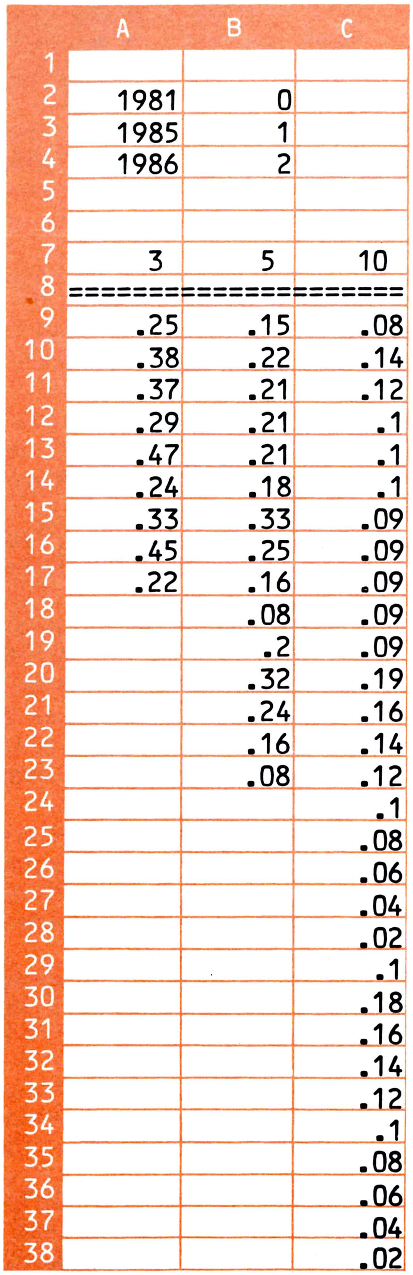
TABLE 1. *ACRS Depreciation Tables (Source: IRS Publication 534).*

To begin your sheet set the column width to seven characters with the entry:

```
/GC7
```

Compare Figure 7 to Table 1. We have rewritten the table by first breaking out the "trigger" years for changes in depreciation rates and creating a small @LOOKUP table in A2...A4 and B2...B4. The purpose of the @LOOKUP table is to assign values to the trigger years. We will use those values to determine where to enter the schedules below. Then we consolidated the rates themselves into three tables. (To save space, we have not included the 15-year schedule in our table: it could be included easily.)

Now enter the @LOOKUP and depreciation tables shown in Figure 7.

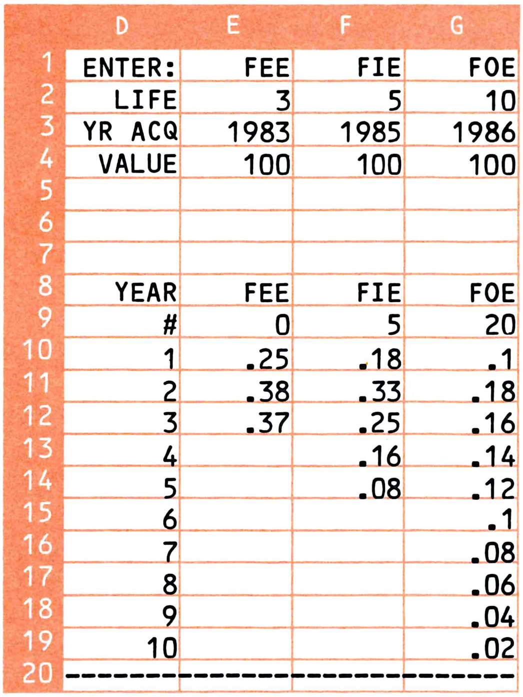


	A	B	C
1			
2	1981	0	
3	1985	1	
4	1986	2	
5			
6			
7	3	5	10
8	=======	=======	=======
9	.25	.15	.08
10	.38	.22	.14
11	.37	.21	.12
12	.29	.21	.1
13	.47	.21	.1
14	.24	.18	.1
15	.33	.33	.09
16	.45	.25	.09
17	.22	.16	.09
18		.08	.09
19		.2	.09
20		.32	.19
21		.24	.16
22		.16	.14
23		.08	.12
24			.1
25			.08
26			.06
27			.04
28			.02
29			.1
30			.18
31			.16
32			.14
33			.12
34			.1
35			.08
36			.06
37			.04
38			.02

FIGURE 7. *Table 1 as adapted for use in the VisiCalc program.*

Next, enter the labels shown in Figure 8 along rows 1 and 8 and in D2...D4. (You may wish to use more realistic names for the properties than Fee, Fie, and Foe.)

After entering values along rows 2...4, as shown in the figure, enter the values 1...10 in column D under the label YEAR #.



	D	E	F	G
1	ENTER:	FEE	FIE	FOE
2	LIFE	3	5	10
3	YR ACQ	1983	1985	1986
4	VALUE	100	100	100
5				
6				
7				
8	YEAR	FEE	FIE	FOE
9	#	0	5	20
10	1	.25	.18	.1
11	2	.38	.33	.18
12	3	.37	.25	.16
13	4		.16	.14
14	5		.08	.12
15	6			.1
16	7			.08
17	8			.06
18	9			.04
19	10			.02
20	-------	-------	-------	-------

FIGURE 8. *Rows five through eight contain no expressions or data. Later, we will split the window and scroll most of what is shown in this figure out of sight.*

To use the consolidated table, we must determine which of its entries apply to a given piece of property. If, for example, we want to depreciate a 5-year property that is put in service in 1985 then we must extract the sequence .18 ... .08. As we have constructed the table, that sequence is in the 6th through the 10th positions of the sequence beginning at B9. For a 3-year property put into service after 1985, we must extract the sequence .33 ... .22, which occupies the 7th through the 9th positions in the sequence beginning in A9.

Now, by looking up the values assigned to the trigger years and multiplying them by the life of the property, we can generate a key that we can use to enter the schedules in the right places. Enter:

```
>E9:@LOOKUP(E3,A1...A4)*E2
```

Replicate that expression into the columns for 5- and 10-year properties with the entry:

```
>E9:/R:F9...G9:RNNR
```

If you entered the years shown in Figure 8 you will have the series 0, 5, 20 in E9...G9. Using those values in an @CHOOSE statement will select the zeroeth, fifth, and twentieth positions in the range or list searched. Using the consolidated table we entered, though, what are needed, are the first, sixth, and twenty-first positions. Those positions in the consolidated tables can be chosen by adding the values at E9...G9 to D10 in the YEAR # column.

That addition specifies the points at which the depreciation sequences begin for the years given in E3...G3. To specify the remaining points in the full sequence, we have only to add the values at E9...G9 to those at D11...D19. The results of those additions can then be chosen from the appropriate columns to the left. Thus, to select the depreciation schedule for a three year property put in service in 1983, enter:

```
>E10:@CHOOSE(E9+D10,A9...A17)
>E10:/R:E11...E12:NRNN
```

At the risk of belaboring the obvious, it's worth pointing out that we are choosing expressions, not single values. The @CHOOSE (like @LOOKUP) function will accept, as its first argument, a numerical value, a reference to a numerical value, or an expression that results in a numerical value. You can also include a function in the first argument. For example, to make the expression at E9 a part of the one at E10, we would have written @CHOOSE(@LOOKUP(E3,A2 . . . A4) *E2+D10,A9 . . . A17). And although the @CHOOSE function will not accept a Boolean value (@TRUE or @FALSE) as its first argument, you can include the Boolean functions of the VisiCalc program within the @IF function and use them to determine the values to be chosen.

Now, to see how this approach works, return to E3 and try various years, watching how the entries at E9...E12 reevaluate.

To create the tables for 5- and 10-year classes of property, enter and replicate:

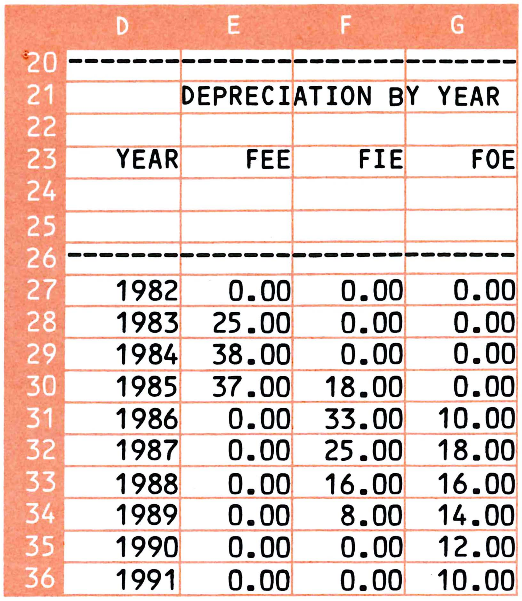
```
>F10:@CHOOSE(F9 +D10,B9...B23)®
    /R:F11...F14:NRNN
>G10:@CHOOSE(G9 +D10,C9...C38)®
    /R:G11...G19:NRNN
```

Those entries produce tables of percentages. As an alternative you can produce a depreciation schedule in dollars by using the expressions shown and multiplying them by the values entered on row 4. Thus, the entry at E10 would be /F$@CHOOSE(E9 +D10,A9 . . . A17)*E4. The entries at F10 and G10 would be multiplied by F4 and G4 respectively. When replicated to fill out the columns, the value references to row 4 would be replicated with No Change (N).

We have chosen to use the tables of percentages, though, and to create yet another table into which we will distribute the amount of depreciation in dollars on a year by year basis.

Distributing Amounts into Year by Year Tables

In general terms the problem is how to write an expression that will allow you to create a table like the one shown in Figure 9. If no depreciation is allowed in a particular year, the program displays 0.00 (formatted with /F$). If, however, depreciation is allowed in a particular year, the appropriate amount is displayed.



	D	E	F	G
20	--------	--------	--------	--------
21		DEPRECI	ATION B	Y YEAR
22				
23	YEAR	FEE	FIE	FOE
24				
25				
26	--------	--------	--------	--------
27	1982	0.00	0.00	0.00
28	1983	25.00	0.00	0.00
29	1984	38.00	0.00	0.00
30	1985	37.00	18.00	0.00
31	1986	0.00	33.00	10.00
32	1987	0.00	25.00	18.00
33	1988	0.00	16.00	16.00
34	1989	0.00	8.00	14.00
35	1990	0.00	0.00	12.00
36	1991	0.00	0.00	10.00

FIGURE 9. *This portion of the sheet makes use of @IF to distribute the allowable depreciation into different years. Notice the difference between this sheet and the one shown in Figure 13.*

The expression we have written to create the table uses the logical function @AND within an @IF statement.

For those who haven't used @IF or @AND or who wish a quick review of the way they work, read on. For those who are on friendly terms with those functions, skip the next three paragraphs.

The @IF function says: IF A is True, THEN give B, ELSE give C. Thus if A is not true, the @IF function will give you C.

The @AND function makes two or more comparisons and if either is false returns the value @FALSE. If both of the comparisons are true, @AND returns @TRUE. (When displaying the values @TRUE or @FALSE, the VisiCalc program shows only TRUE or FALSE on the screen. The @ symbol is implicit in the display, however.)

When @IF contains @AND, and @AND evaluates to TRUE, that is, all of the comparisons are true, the @IF statement will give B. But if the @AND evaluates to FALSE, because at least one of the comparisons is false, the @IF statement will give you C.

Begin this part of the sheet by entering the label "1st Year" at D5 and the labels from D20 to G26. Enter the years in column D. Then enter:

```
>E5:1982
>D27:+E5
>D28:+D27+1®/R:D29...D36:R
```

To produce 0.00 for any year in which no depreciation is allowed, we will let @AND evaluate to FALSE, meaning that the function must include at least one false comparison.

Rather than entering the entire expression needed to produce the table, we shall look at it piece by piece.

Enter:

```
>E27:+E3<=D27
>F27:+E2+E3>D27
```

Since E3 is 1983, E2 is 3, and E5 is 1982, those two statements will return FALSE and TRUE. Now enter:

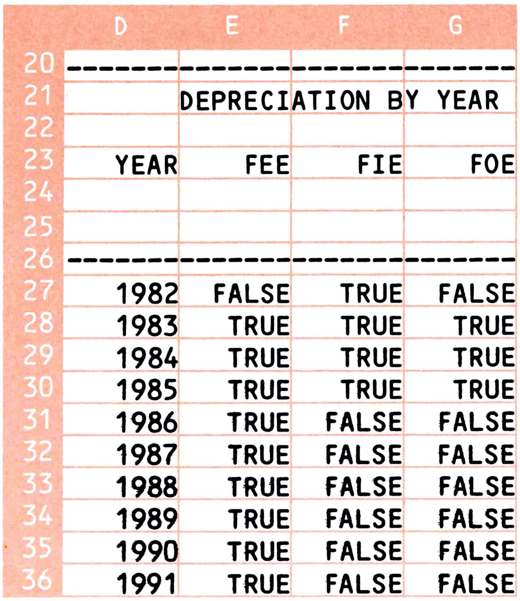
```
>G27:@AND(E27,F27)
```

and

```
>E27:/R...G27:E28...E36:NRNNRRR
```

The test in column E (whether the value at E3 is less than or equal to the corresponding value in column D) defines the earliest year in the allowable sequence, while the test in column F defines the latest year. Neither test could define both limits, so we use @AND. The actual expression that defines the earliest and latest years (or months if you are constructing a monthly table) is @AND(E3< =D27,E2+E3>D27), which is a compressed version of the expressions now on your sheet at E27, F27, and G27.

Your sheet should now resemble Figure 10. Change the year shown at D27 and you will see how the @AND statement evaluates down the column. If you change the year at D27, by the way, remember to change it back to +E5 before making the next entry below.



	D	E	F	G
20	--------	--------	--------	--------
21		DEPRECIATION BY YEAR		
22				
23	YEAR	FEE	FIE	FOE
24				
25				
26	--------	--------	--------	--------
27	1982	FALSE	TRUE	FALSE
28	1983	TRUE	TRUE	TRUE
29	1984	TRUE	TRUE	TRUE
30	1985	TRUE	TRUE	TRUE
31	1986	TRUE	FALSE	FALSE
32	1987	TRUE	FALSE	FALSE
33	1988	TRUE	FALSE	FALSE
34	1989	TRUE	FALSE	FALSE
35	1990	TRUE	FALSE	FALSE
36	1991	TRUE	FALSE	FALSE

FIGURE 10. *The sheet in this figure shows how the limits are defined for the distribution of values into the year by year table.*

The next step is to choose from the table above the depreciation rate that corresponds to the year in column D. Enter:

```
>H27:1+D27–E3®/R:H28...H36:RN
```

Notice that the sequence TRUE, TRUE, TRUE in column G matches the sequence 1, 2, 3 in column H. If we make the expression we just entered a part of @CHOOSE and make the @CHOOSE a part of an @IF containing the @AND function that produced those TRUEs, we have the basic expression that produced the table in Figure 9. Enter:

```
>E27:/F$@IF(@AND(E3<=D27,
    E2+E3>D27), @CHOOSE(D27–E3+1,
    E10...E19)*E4,0)
```

Replicate that expression into columns F and G, overwriting the entries shown above (and erase H27...H36 with a separate replication of a blank cell). Enter:

```
>E27:/R:F27...G27:RNRRNNRRRR
```

Then, repeating the value references three times, fill the columns with the remaining calculations by entering:

```
>E27:/R...G27:E28...E36:NRNNRRNNNN
```

Once you have completed those entries you'll have the foundation for analyzing properties according to ACRS. Depreciation, of course, is only one factor in analyzing costs. There are probably other factors (Investment Tax Credits, for example) that you might want to add to the model. In any event, you are likely to be dealing with more than one property in the same class, so you will have to add more columns to the sheet. That, and how to format the sheet for use by an inexperienced user will conclude our discussion.

Expanding and Managing the Model

To write a new column for analyzing another piece of property calls for inserting a column and replicating the appropriate schedule into the blank space. Rather than replicate an entire column, which would entail entering a large number of value references, it is easier to break the column into two pieces, replicating across first and then down. Enter:

```
>F9:/IC
>E9:/R...E10:F9:RNNRRNNN
>F10:/R:F11...F12:NRNN
```

The same procedure is followed below when replicating the year by year table. Enter:

```
>E27:/R:F27:RNRRNNRRRR
```

and

```
>F27:/R:F28...F36:NRNNRRNNNN
```

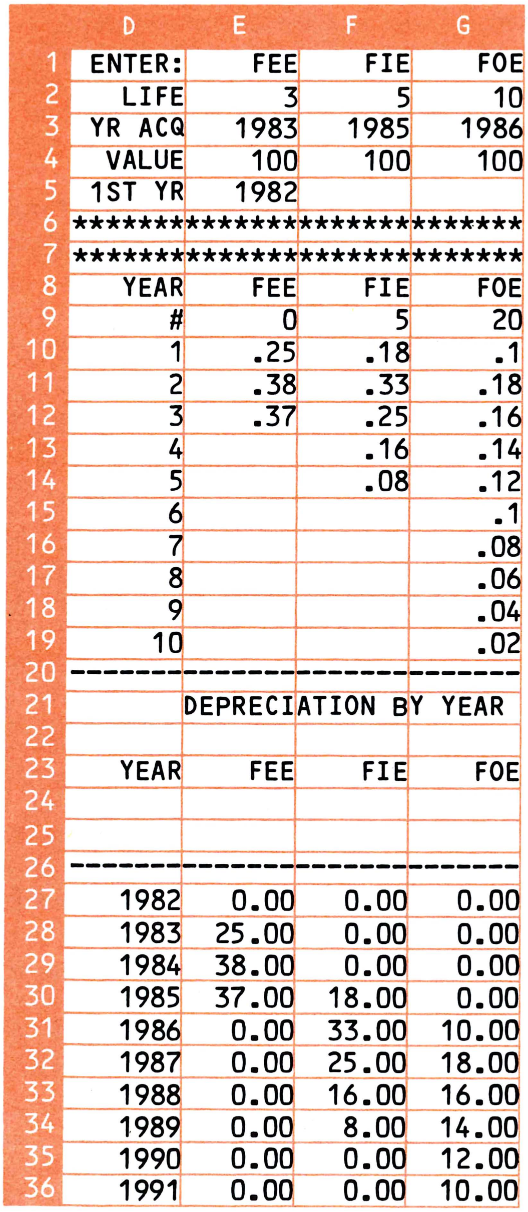
HAVE YOU MOVED?

If you have, let us know. To keep your issues of SATN coming to you regularly, we need to know your current address. Send a card or letter, indicating both your old address and your new one, to:

SATN Subscriptions
P.O. Box 815
Quincy, MA 02169

Remember to allow about 4 weeks for the change to take effect.

Once the basic expressions have been entered as shown, the sheet can be set up for use by an inexperienced user, a data entry clerk. Figures 11 and 12 show a sheet for three depreciable items set up this way.

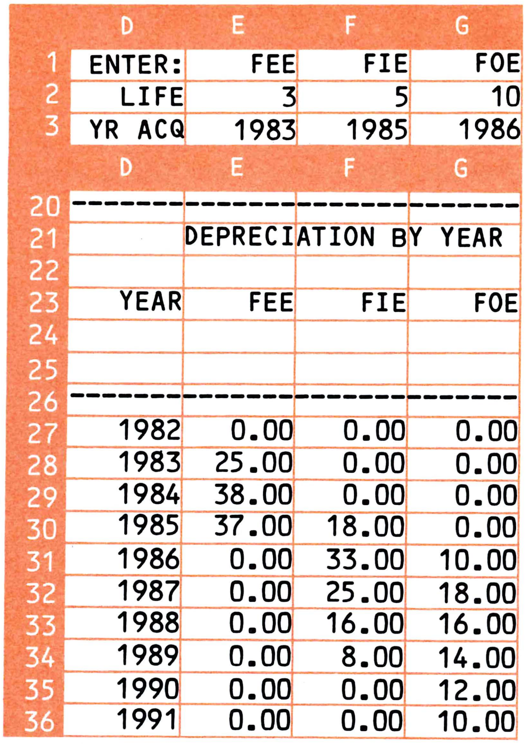


	D	E	F	G
1	ENTER:	FEE	FIE	FOE
2	LIFE	3	5	10
3	YR ACQ	1983	1985	1986
4	VALUE	100	100	100
5	1ST YR	1982		
6	**********	**********	**********	**********
7	**********	**********	**********	**********
8	YEAR	FEE	FIE	FOE
9	#	0	5	20
10	1	.25	.18	.1
11	2	.38	.33	.18
12	3	.37	.25	.16
13	4		.16	.14
14	5		.08	.12
15	6			.1
16	7			.08
17	8			.06
18	9			.04
19	10			.02
20	----------	----------	----------	----------
21		DEPRECIATION BY YEAR		
22				
23	YEAR	FEE	FIE	FOE
24				
25				
26	----------	----------	----------	----------
27	1982	0.00	0.00	0.00
28	1983	25.00	0.00	0.00
29	1984	38.00	0.00	0.00
30	1985	37.00	18.00	0.00
31	1986	0.00	33.00	10.00
32	1987	0.00	25.00	18.00
33	1988	0.00	16.00	16.00
34	1989	0.00	8.00	14.00
35	1990	0.00	0.00	12.00
36	1991	0.00	0.00	10.00

FIGURE 11. *The sheet before the /W command is given. The repeating asterisks (produced with /-*) provide a warning track of sorts, to guard against entering data in the incorrect locations.*

The entries to set up the windows shown, and set title boundaries, are:

```
>D4:/WH
>D3:/TB;
>D6:/-*®/R:E6...J6:/R...J6:D7
>D35:>D23:/TB
```



	D	E	F	G
1	ENTER:	FEE	FIE	FOE
2	LIFE	3	5	10
3	YR ACQ	1983	1985	1986
	D	E	F	G
20	----------	----------	----------	----------
21		DEPRECIATION BY YEAR		
22				
23	YEAR	FEE	FIE	FOE
24				
25				
26	----------	----------	----------	----------
27	1982	0.00	0.00	0.00
28	1983	25.00	0.00	0.00
29	1984	38.00	0.00	0.00
30	1985	37.00	18.00	0.00
31	1986	0.00	33.00	10.00
32	1987	0.00	25.00	18.00
33	1988	0.00	16.00	16.00
34	1989	0.00	8.00	14.00
35	1990	0.00	0.00	12.00
36	1991	0.00	0.00	10.00

FIGURE 12. *The sheet as set up for final use. If @NA is entered at E5, the year by year table will consist entirely of @NA.*

The title boundaries and the repeating rows of asterisks serve as warnings to a data entry clerk. You would, of course, create the template with as many columns as necessary. By entering @NA at E5, all of the calculations in the lower window will display as NA.

Rows six and seven have been left effectively blank, that is, they contain no tables or calculations, so that we can insert additional rows on which instruction labels can be entered for the guidance of untutored users. Among the instructions that might be entered, for example, would be the ones necessary to create a printed report, including the appropriate printer string.

As we set it up, the expression used to create the year by year table returns either something or nothing, that is, 0. Given our example, that is appropriate. There are many other problems, however, for which the same basic technique can be used, but which would require an alternative other than zero.

In closing, we'd like to say once more that although we have emphasized ACRS depreciation, there are many other problems for which you can use the techniques shown in this article. For example, the year by year table might be used to track monthly expenses in various categories that do not recur each month, or for any problem in which some, but not all, elements overlap. And there's no reason why the key to the table must be years or months. The comparison could be between regions and income accounts. Lastly, review the way that we have used @CHOOSE. We could have used @LOOKUP once we had generated the tables of percentages, but then we would have had to add index ranges to the sheet. Those index ranges would have used memory. If you foresee bumping into your computer's memory limits, consider substituting @CHOOSE for @LOOKUP.

A LOOK AT VISICALC ADVANCED VERSION

We expect only a few of you are using the VisiCalc Advanced Version program as yet, so we will not go into detail now concerning its many new features. However, there are several aspects of the new /A (Attributes) command that we have highlighted in Figure 13, which is a screen photo of the sheet developed for ACRS Depreciation.

Among the things for which the Attributes command is used are several globally or locally defined elements in the display of your VisiCalc Advanced Version sheet. Those we have used on the sheet in Figure 13 include /AL, /AV, and /AV% for the display of labels, values, and percentages.

The labels on rows 1, 8, and 23 are neither flush left (the default), nor flush right, but are centered in the columns, even though the columns are of varying widths. Although it has always been possible to center labels by adding leading blanks to the label entry, it can now be done with /ALC. If you change the width of a column, labels that have been set with /ALC will remain centered. Being able to center labels easily should enable you to produce good looking reports in less time.

Value displays are highly flexible in VisiCalc Advanced Version. To display 1000 as $1,000.00 we entered:

```
>E4:/AV,$2F
```

The /AV portion of that entry specifies that we wished to set the attributes for the display of a value. The comma is entered to indicate that we wish commas to appear in the appropriate locations in all values of 1000 or more. The dollar sign produces the display of that symbol. Finally, the "2F" portion of the entry specifies that we want to "fix" two decimal places. The display can be set to as many decimal places as you need (up to approximately 125).

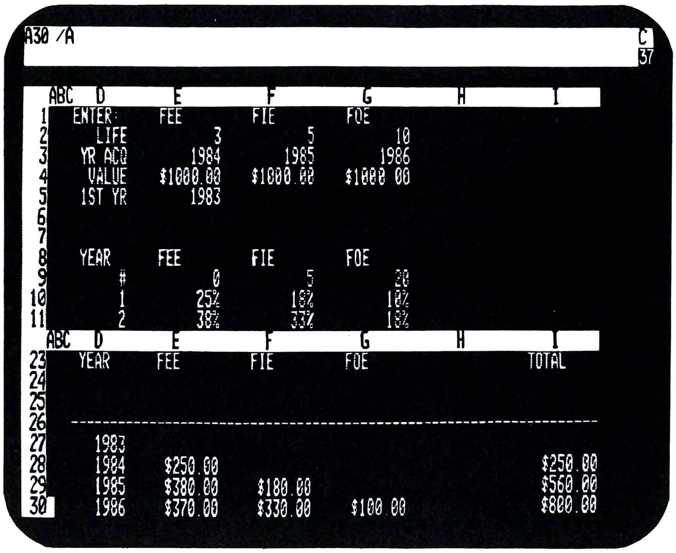
In the VisiCalc Advanced Version program you can replicate an entry as you do in the VisiCalc program, but you can also replicate the Attributes only. That is, if you have entered a value or expression in a cell and wish to change the way it's displayed, you can replicate the Attributes from another cell, without affecting the value or expression that you have already entered in the target cell or cells. We had already entered 1000 at F4 and G4, so we replicated the Attributes at E4 with the entry:

```
>E4:/R(A):F4...G4
```

We also used the Attributes command to display the percentages shown in rows 10 and 11. In the VisiCalc Advanced Version program, if a cell is formatted with the command /AV% it will display a value that is 100 times greater than the value that has been entered. Thus, when .25 is entered in a cell that has been set with /AV%, it will display at 25%. That is, we entered:

```
>E10:/AV%®/R(A):F10...G10
```

We then replicated the Attributes in E10 . . . G10 into the cells below with:



```
A30 /A                                                          C
                                                                37
  ABC  D          E          F          G          H          I
   1 ENTER:      FEE        FIE        FOE
   2    LIFE          3          5         10
   3  YR ACQ       1984       1985       1986
   4   VALUE   $1000.00   $1000.00   $1000.00
   5  1ST YR       1983
   6
   7
   8   YEAR      FEE        FIE        FOE
   9       #          0          5         20
  10       1        25%        18%        10%
  11       2        38%        33%        18%
  ABC  D          E          F          G          H          I
  23   YEAR      FEE        FIE        FOE                   TOTAL
  24
  25
  26 ---------------------------------------------------------------
  27    1983
  28    1984   $250.00                                      $250.00
  29    1985   $380.00    $180.00                           $560.00
  30    1986   $370.00    $330.00    $100.00                $800.00
```

FIGURE 13. *This photo shows the VisiCalc Advanced Version screen. Notice that columns A, B, and C are only one character wide, and appear to be empty. In fact, the ACRS depreciation tables that are entered in those columns have been hidden from view with the /AH (Hide) command, which can be used to hide intermediate calculations or sensitive data from view.* *Photo by Sonny Tarbi.*

```
>E10:/R(A)...G10:E11...E19
```

On the subject of dealing with percentages in VisiCalc Advanced Version: if you enter the value 25 and then press the % key (shift 5), the program will automatically enter .25 in memory. Given that, and /AV%, it should be unnecessary in most instances to multiply or divide by 100 to manipulate percentages.

Notice that there are no 0.00 entries in the year by year table. Wherever a zero dollar amount would appear in the VisiCalc program, we now show a blank. To produce that display we substituted @LABEL for 0 in the @IF expression used to produce the table. That is, the entry:

```
>E27:@IF(@AND(E3<=D27,E2+E3>D27),
@CHOOSE(D27-E3+1,E10...E19)*E4,0)
```

as shown in ACRS Depreciation, has been altered to read:

```
>E27:@IF(@AND(E3<=D27,E2+E3>D27),
@CHOOSE(D27-E3+1,E10...E19)
*E4,@LABEL(A1))
```

By leaving A1 blank and protecting it with /M (Mode), which we'll look at in more detail in future issues, we can prevent any entries whatsoever from being made there and so can always use @LABEL(A1) to return a blank cell whenever it is appropriate. Of course, using @LABEL or @LCHOOSE (for Label Choose), we can also return a label that may be at some location.

In general, the VisiCalc Advanced Version program will allow you to generate reports with surprising ease. In forthcoming issues of SATN we shall gradually introduce VisiCalc Advanced Version's many powerful new features.

VISICALC RELATED PUBLICATIONS

The first criterion for including any article, book, or newsletter in this listing was whether we could find a hard copy in our library. The second was that the item appeared to be useful. We have included only articles that contain information that can work for you, either in the form of models that you can use or reviews of various products that complement the VisiCalc® program.

Whether the information, expressions, or advice in any of the items included on the list are accurate has not been verified by anyone at Software Arts, Inc. That is, we have neither reviewed nor examined any of these books, articles, or publications to determine their accuracy or freedom from error. Inclusion in this listing does not constitute an endorsement by Software Arts, Inc. of any such book, article, or publication.

Books

Harry Anbarlian, *An Introduction to VisiCalc Matrixing for the Apple and IBM*, McGraw-Hill, 1982. This book consists of many VisiCalc templates and applications, covering a wide range of subjects.

Donald Beil, *The VisiCalc Book, Apple Edition*, Reston Publishing Co., 1982. A book giving a full explanation of VisiCalc on the Apple computer, from very basic "what is VisiCalc" to machine-specific information, tutorials and practice problems, and lists of common errors.

Journals and Newsletters

AgriComp: The Reference for Farm Computing, 1001 E. Walnut, Suite 201, Columbia, MO 65201

This is a bimonthly magazine aimed at farmers. VisiCalc related articles in the premier (Jan/Feb 1982) issue include: "What is VisiCalc Anyway?"; "Visicalibrate, Using VC to calibrate sprayer nozzles"; "Planning Budget, a VC template for cash flow".

Spreadsheet, VisiGroup, PO Box 254, Scarsdale, NY 10583

This is the VisiGroup (The International VisiCalc Users' Group) newsletter, begun in November 1980. It contains news and applications useful to VisiCalc users.

Regular Columns

"Visulating", by Barry Bayer, *Desktop Computing*

November 1981: The initial column describes VisiCalc and develops an amortization table as an example of how to use VisiCalc.
April 1982: This column uses VisiCalc to do averages, and includes grids for keeping track of school grades and baseball players' statistics.

"Ventures with VisiCalc", *Softalk 1*

March 1981, by Frank Malone: The initial column introduces VisiCalc.
April 1981, by Frank Malone: Family finances, including checkbook and income and expense records are done using VisiCalc.
June 1981, by Otto K. Wetzel: Wetzel correlates and plots utility bills using weather data.
August 1981, by Craig Stinson: This column reviews VisiCalc on the Apple /// and comments on some products that work with VisiCalc but are not published by VisiCorp: V-Plot, V-Stat, V-Print, Visiprint, and Visi-Caids.
February 1982, by Craig Stinson: This shows how to do interest calculations using VisiCalc.
March 1982, by Craig Stinson: Stinson sets up income tax calculations on VisiCalc.
April 1982, by Joe Shelton: This column contains explanations of VisiCalc's built in functions.

"Apple Visi Corner", by Jim Englander *NIBBLE*

Vol2 No6 1981: How to do "greater than" and "less than" with versions of VisiCalc before 193B0. (That is, before DOS 3.3.)
Vol3 No2 1982: Englander shows how to do calendar control and business reporting using VisiCalc.

Articles

Successful Farming, July 1981, "Using VisiCalc: Building a cash flow chart". This article leads you through preparing a cash flow statement, giving formulas and possible categories to use.

Creative Computing, May 1981. "Break Even Analysis with VisiCalc", by George Blank. These VisiCalc formulas calculate return on various quantities of goods sold, to determine break-even points.

INC, November 1981. "How VisiCalc Works", by Stewart Alsop II. This is an explanation of VisiCalc for people who are not versed in computers.

TRS-80 Microcomputer News, May, 1982. "Divying Up", This is an article on budget planning using VisiCalc.

LOOKING AHEAD

This issue of SATN begins Volume II. Some topics for future issues include using VisiCalc with selected word processing, database, and plotting programs, and more on using @AND, @OR, and @IF.

If your subscription began with Volume I, Number 1, it is time to think about renewing.

The cost of your regular SATN subscription hasn't changed. Six issues are still $30.00 in the US, or $50.00 outside the US. As a special offer, you can order 12 issues for a reduced price: $50.00 in the US, or $90.00 outside the US.

Send your subscription order and check, MasterCard, or VISA information to:

SATN Subscriptions
PO Box 815
Quincy, MA 02169

Don't let your advanced lessons in VisiCalc fall by the wayside. Act now!